

APPENDIX A
Least Square Lattice
Noise Cancelling

/* Example for ratiometric approach to noise cancelling */
#define LAMBDA 0.95

```
void OxLSL_NC( int    reset,
               int    passes,
               int    *signal_1,
               int    *signal_2,
               int    *signal_3,
               int    *target_1,
               int    *target_2) {

    int    i, ii, k, m, n, contraction;
static int    *s_a, *s_b, *s_c, *out_a, *out_c;
static float  Delta_sqr, scale, noise_ref;
```

```
if( reset == TRUE){
    s_a = signal_1;
    s_b = signal_2;
    s_c = signal_3;
    out_a = target_1;
    out_c = target_2;
    factor = 1.5;
    scale = 1.0 / 4160.0;

    * noise canceller initialization at time t=0 */
```

```
nc[0].berr = 0.0;
nc[0].Gamma = 1.0;

for(m=0; m<NC_CELLS; m++) {
    nc[m].err_a = 0.0;
    nc[m].err_b = 0.0;
    nc[m].Roh_a = 0.0;
    nc[m].Roh_c = 0.0;
    nc[m].Delta = 0.0;
    nc[m].Fwsqr = 0.00001;
    nc[m].Bwsqr = 0.00001;
}
```

/*===== END INITIALIZATION =====*/

```
for(k=0; k<passes; k++){
    contraction = FALSE;
    for(m=0; m< NC_CELLS; m++) {
        nc[m].berr1 = nc[m].berr;
        nc[m].Bwsqr1 = nc[m].Bwsqr;
    }

    noise_ref = factor * log(1.0 - (*s_a) * scale)
                - log(1.0 - (*s_b) * scale);
    nc[0].err_a = log(1.0 - (*s_b) * scale);
    nc[0].err_b = log(1.0 - (*s_c) * scale);
```

Figure 1 consists of seven sub-graphs, labeled (a) through (g), each representing a different fish species. The x-axis for all graphs is 'Time of day' in 24-hour format (0000 to 2400). The y-axis is 'Percentage of total catch'. The data is plotted as a line graph with error bars.

- (a) Yellowtail snapper: Shows a peak in the morning (around 0600) and another peak in the evening (around 1800).
- (b) Red snapper: Shows a peak in the morning (around 0600) and a smaller peak in the evening (around 1800).
- (c) Grey snapper: Shows a peak in the morning (around 0600) and a smaller peak in the evening (around 1800).
- (d) Black snapper: Shows a peak in the morning (around 0600) and a smaller peak in the evening (around 1800).
- (e) White snapper: Shows a peak in the morning (around 0600) and a smaller peak in the evening (around 1800).
- (f) Blue snapper: Shows a peak in the morning (around 0600) and a smaller peak in the evening (around 1800).
- (g) Silver snapper: Shows a peak in the morning (around 0600) and a smaller peak in the evening (around 1800).

```
/* Order Update      */
for(n=1; ( n < NC CELLS) && (contraction == FALSE); n++) {
```

```
m = n-1;
ii= n-1;
```

```
nc[n].fref = -nc[m].Delta / nc[m].Bswsqrl;  
nc[n].bref = -nc[m].Delta / nc[m].Fswsqr;
```

```
nc[n].Fwsqr = nc[m].Fwsqr - Delta_sqr / nc[m].Bwsqr1;
nc[n].Bwsqr = nc[m].Bwsqr1 - Delta_sqr / nc[m].Fwsqr;
```

/* Joint Process Estimation Section */

```

nc[m].Roh_c  *= LAMBDA;
nc[m].Roh_c  += nc[m].berr * nc[m].err_b / nc[m].Gamma ;
nc[m].k_c    = nc[m].Roh_c / nc[m].BSwsqr;
nc[n].err_b  = nc[m].err_b - nc[m].k_c * nc[m].berr;

```

```

}
else {
    contraction = TRUE;
    for(i=n; i<NC_CELLS; i++) {
        nc[i].err_a = 0.0;
        nc[i].Roh_a = 0.0;
        nc[i].err_b = 0.0;
        nc[i].Roh_c = 0.0;
        nc[i].Delta = 0.0;
        nc[i].Fswsq = 0.00001;
        nc[i].Bswsq = 0.00001;
        nc[i].Bswsq1 = 0.00001;
    }
}

```

```
    }  
  }  
}  
*out_a++ = (int)( (-exp(nc[ii].err_a) +1.0) / scale) ;  
*out_c++ = (int)( (-exp(nc[ii].err_b) +1.0) / scale) ;
```

```
}  
/***** Least Square Lattice *****/  
/*****  
/*****/
```

034434 100697